Massive Database Generation for 2.5D Borehole Electromagnetic Measurements using Refined Isogeometric Analysis

Ali Hashemian[a,*], Daniel Garcia[b,c], Jon Ander Rivera[d,a], David Pardo[d,a,e]

[a] *BCAM – Basque Center for Applied Mathematics, Bilbao, Basque Country, Spain*
[b] *CIMNE – International Center for Numerical Methods in Engineering, Barcelona, Cataluña, Spain*
[c] *IDAEA – Institute of Environmental Assessment and Water Research, Barcelona, Cataluña, Spain*
[d] *University of the Basque Country UPV/EHU, Leioa, Basque Country, Spain*
[e] *Ikerbasque – Basque Foundation for Sciences, Bilbao, Basque Country, Spain*

Abstract

Borehole resistivity measurements are routinely inverted in real-time during geosteering operations. The inversion process can be efficiently performed with the help of advanced artificial intelligence algorithms such as deep learning. These methods require a massive dataset that relates multiple Earth models with the corresponding borehole resistivity measurements. In here, we propose to use an advanced numerical method —refined isogeometric analysis (rIGA)— to perform rapid and accurate 2.5D simulations and generate databases when considering arbitrary 2D Earth models. Numerical results show that we can generate a meaningful synthetic database composed of 100,000 Earth models with the corresponding measurements in 56 hours using a workstation equipped with two CPUs.

Keywords: Geosteering; borehole resistivity measurements; refined isogeometric analysis; 2.5D numerical simulation; deep learning inversion.

1. Introduction

Geosteering plays a crucial role in oil and gas engineering. To perform geosteering operations, companies often employ borehole instruments that record electromagnetic (EM) data in real-time while drilling (Liu, 2017a). Since the electrical resistivity is highly sensitive to salinity, EM measurements are used to distinguish between hydrocarbon- and water-saturated rocks (Liu, 2017b).

Inversion techniques estimate layer-by-layer EM properties from the measurements, allowing for the adjustment of the logging trajectory during geosteering operations. Thus, they enable to select an optimal well trajectory toward the target hydrocarbon-saturated rocks. There exist a plethora of inversion methods in the literature, including gradient based methods (Vogel, 2002; Tarantola, 2005), statistics based methods (Watzenig, 2007; Kaipio and Somersalo,

☆Credit author statement is as follows. Ali Hashemian: Data curation, formal analysis, investigation, validation, visualization, writing—original draft, writing—review & editing. Daniel Garcia: Formal analysis, investigation, methodology, software, writing—original draft, writing—review & editing. Jon Ander Rivera: Investigation, methodology, software; David Pardo: Funding acquisition, project administration, resources, supervision, writing—review & editing.

*Corresponding author
Email address: ahashemian@bcamath.org (Ali Hashemian)

2007; Shen et al., 2020), and artificial intelligence based methods (Kim and Nakata, 2018; Liu and Grana, 2019; Yang and Ma, 2019; Li et al., 2020). In geosteering EM measurements, deep learning (DL) methods with advanced encoder-decoder neural networks have recently demonstrated to be suitable to solve inverse problems (Shahriari et al., 2020a,b).

DL methods are fast, but require a massive training dataset. To decrease the *online* computational time during field operations, we often produce such a large dataset *a priori (offline)* using tens of thousands of simulations of borehole resistivity measurements (see, e.g., Kushnir et al., 2018). To generate the database for DL inversion, we employ simulation methods to solve Maxwell's equations with different conductivity distributions (Earth models). Since 3D simulations are expensive and possibly unaffordable when computing such large databases, it is common to reduce the Earth model dimensionality to two or one spatial dimensions using a Fourier or a Hankel transform. These transformations lead to the so-called 2.5D (Abubakar et al., 2006; Pardo et al., 2008; Shen and Sun, 2008; Nam et al., 2013; Gernez et al., 2020) and 1.5D (Pardo and Torres-Verdín, 2015; Bakr et al., 2017; Shahriari and Pardo, 2020) formulations, respectively. 1.5D simulations are inaccurate when dealing with geological faults. In this work, we focus on the efficient generation of a massive database using 2.5D simulations —as a preliminary stage for deep learning inversions.

Galerkin methods are effective for simulating well-logging problems (see, e.g., Pardo et al., 2006, 2021; Calo et al., 2011; Ma et al., 2012; Wang et al., 2013; Rodríguez-Rozas et al., 2018; Chaumont-Frelet et al., 2018b; Nunes and Régis, 2020). Isogeometric analysis (IGA), introduced by Hughes et al. (2005), is a widely used Galerkin method for solving partial differential equations. IGA has been successfully employed in various electromagnetic (Buffa et al., 2010, 2014; Nguyen et al., 2012; Simpson et al., 2018; Simona et al., 2020) and geotechnical (Shahrokhabadi et al., 2019; Hageman et al., 2019) applications. IGA uses spline basis functions introduced in computer-aided design (CAD) as shape functions of finite element analysis (FEA). These basis functions exhibit high continuity (up to C^{p-1}, being p the polynomial order of spline bases) across the element interfaces.

When comparing IGA and FEA, the former provides smoother solutions for wave propagation problems with a lower number of unknowns (Hughes et al., 2005; Cottrell et al., 2009). However, in contrast to the minimal interconnection of elements in FEA, high-continuity IGA discretizations strengthen the interconnection between elements, leading to an increase of the cost of matrix LU factorization per degree of freedom when using sparse direct solvers (Collier et al., 2012). In order to avoid this degradation and also benefit from the recursive partitioning capability of multifrontal direct solvers, Garcia et al. (2017) developed a new method called refined isogeometric analysis (rIGA). This discretization technique conserves desirable properties of high-continuity IGA discretizations, while it partitions the computational domain into blocks of macroelements weakly interconnected by low-continuity separators. As a result, the computational cost required for performing LU factorization decreases. The applicability of the rIGA framework to general EM problems was studied by Garcia et al. (2019). Compared to high-continuity IGA, rIGA produces solutions of EM problems up to $\mathcal{O}(p^2)$ faster on large domains and close to $\mathcal{O}(p)$ faster on small domains. rIGA also improves the approximation errors with respect to IGA since the continuity reduction of basis

functions increases the number of degrees of freedom (c.f. Cottrell et al., 2009) and enriches the Galerkin space.

Herein, we propose the use of rIGA discretizations to generate databases for DL inversion of 2.5D geosteering EM measurements. We consider *a priori* grids following the idea of optimal grid generation for 2.5D EM measurements presented by Rodríguez-Rozas et al. (2018) and the methods described by Rodríguez-Rozas and Pardo (2016) for Fourier mode selections. Compared to the FEA approach described by Rodríguez-Rozas et al. (2018) that assigns increasing polynomial orders for the elements near the well, we consider a (smooth) high-continuity IGA discretization with a fixed polynomial order everywhere and reduce the computational cost by continuity reduction of certain basis functions in the sense of rIGA framework. To assess the accuracy and computational efficiency of the rIGA approach in borehole resistivity simulations, we consider several model problems where high-angle wells cross spatially heterogeneous media exhibiting multiple geological faults. Then, we investigate the performance of the proposed approach when generating a synthetic database composed of 100,000 Earth models.

The remainder of this article is organized as follows. In Section 2, we review the governing equations of the 3D EM wave propagation problem and derive the 2.5D variational formulation. Section 3 introduces the considered borehole resistivity problem. In Section 4, we describe both high-continuity and refined isogeometric discretizations, followed by the implementation details in Section 5. We analyze the accuracy and computational efficiency of the rIGA approach when applied to borehole resistivity problems in Section 6. In this section, we also generate a database for DL inversion of geosteering measurements. Section 7 presents some discussions about the results and Section 8 draws the main conclusions and possible future research lines stemming from this work.

2. 2.5D variational formulation of EM measurements

2.1. 3D wave propagation problem

The two time-harmonic curl Maxwell's equations describing the 3D wave propagation in an isotropic medium are

$$\nabla \times \mathbf{E} + \mathrm{i}\omega\mu\mathbf{H} = -\mathrm{i}\omega\mu\mathbf{M}\,, \tag{1}$$

$$\nabla \times \mathbf{H} = (\sigma + \mathrm{i}\omega\varepsilon)\mathbf{E}\,, \tag{2}$$

where $\mathbf{E}$ is the electric field, $\mathbf{H}$ is the magnetic field, i is the imaginary unit, σ is the electric conductivity, ε is the electric permittivity, μ is the magnetic permeability, $\omega = 2\pi f$ is the angular frequency, with f being the transmitter frequency, and $\mathbf{M}$ is the time-harmonic magnetic source located at (x_0, y_0, z_0) and given by

$$\mathbf{M} = \delta(x - x_0)\delta(y - y_0)\delta(z - z_0)[M_x\,, M_y\,, M_z] \qquad \text{in } \mathbb{R}^3, \tag{3}$$

with $\delta(\cdot)$ being the Dirac delta function defined as follows:

$$\delta(x - x_0) := \begin{cases} \infty\,, & x = x_0\,, \\ 0\,, & x \neq x_0\,. \end{cases} \tag{4}$$

To incorporate an integrable approximation of the Dirac delta function, we consider a bell-like representation for the delta function. For example, in the x direction, we approximate:

$$\delta(x - x_0) \approx \frac{1}{\alpha\sqrt{\pi}} \exp\left[-\left(\frac{x - x_0}{\alpha}\right)^2\right], \tag{5}$$

where α is a positive value.

From Maxwell's equations, we obtain the following reduced wave formulation in terms of magnetic field $\mathbf{H}$:

$$\begin{cases} \text{Find } \mathbf{H} = [\mathrm{H}_x\,, \mathrm{H}_y\,, \mathrm{H}_z], \text{ with } \mathbf{H} : \boldsymbol{\Omega} \subset \mathbb{R}^3 \to \mathbb{C}^3, \text{ such that:} \\ \qquad \nabla \times \left(\dfrac{1}{\sigma + \mathrm{i}\omega\varepsilon} \nabla \times \mathbf{H}\right) + \mathrm{i}\omega\mu\mathbf{H} = -\mathrm{i}\omega\mu\mathbf{M} \qquad \text{in } \boldsymbol{\Omega}\,, \\ \qquad\qquad\qquad\qquad \mathbf{H} \times \mathbf{n} = \mathbf{0} \qquad\qquad \text{on } \partial\boldsymbol{\Omega}\,, \end{cases} \tag{6}$$

where $\boldsymbol{\Omega}$ is the domain of study and $\mathbf{n}$ is the unit normal (outward) vector on the boundary $\partial\boldsymbol{\Omega}$. We define $\boldsymbol{\Omega}$ as a tensor-product box:

$$\begin{aligned} \boldsymbol{\Omega} &= \Omega_x \times \Omega_y \times \Omega_z \\ &= (-L_x/2, L_x/2) \times (-L_y/2, L_y/2) \times (-L_z/2, L_z/2)\,, \end{aligned} \tag{7}$$

being L_x,L_y, and L_z positive real constants.

To introduce the weak formulation of this problem, we first define the $\boldsymbol{H}(\mathrm{curl};\boldsymbol{\Omega})$-conforming functional spaces

$$\boldsymbol{H}(\mathrm{curl};\boldsymbol{\Omega}) := \left\{\mathbf{W} = [\mathrm{W}_x\,, \mathrm{W}_y\,, \mathrm{W}_z] \in (L^2(\boldsymbol{\Omega}))^3 : \nabla \times \mathbf{W} \in (L^2(\boldsymbol{\Omega}))^3\right\}, \tag{8}$$

$$\boldsymbol{H}_0(\mathrm{curl};\boldsymbol{\Omega}) := \left\{\mathbf{W} \in \boldsymbol{H}(\mathrm{curl};\boldsymbol{\Omega}) : \mathbf{W} \times \mathbf{n} = \mathbf{0} \text{ on } \partial\boldsymbol{\Omega}\right\}. \tag{9}$$

The $\boldsymbol{H}(\mathrm{curl};\boldsymbol{\Omega})$ space is endowed with the inner product

$$\begin{aligned} (\mathbf{W}, \mathbf{H})_{\boldsymbol{H}(\mathrm{curl};\boldsymbol{\Omega})} &:= (\nabla \times \mathbf{W}, \nabla \times \mathbf{H})_{(L^2(\boldsymbol{\Omega}))^3} + (\mathbf{W}, \mathbf{H})_{(L^2(\boldsymbol{\Omega}))^3} \\ &:= \int_{\boldsymbol{\Omega}} (\nabla \times \mathbf{W})^* \cdot (\nabla \times \mathbf{H})\, d\boldsymbol{\Omega} + \int_{\boldsymbol{\Omega}} \mathbf{W}^* \cdot \mathbf{H}\, d\boldsymbol{\Omega}\,, \end{aligned} \tag{10}$$

where * is the conjugate transpose of complex vector space and $\cdot$ denotes the inner product.

We build the weak formulation by multiplying Eq. (6) with an arbitrary function $\mathbf{W} \in \boldsymbol{H}_0(\mathrm{curl};\boldsymbol{\Omega})$, using Green's formula, and integrating over the domain $\boldsymbol{\Omega}$. The weak formulation is then

$$\begin{cases} \text{Find } \mathbf{H} \in \boldsymbol{H}_0(\mathrm{curl};\boldsymbol{\Omega}), \text{ such that for every } \mathbf{W} \in \boldsymbol{H}_0(\mathrm{curl};\boldsymbol{\Omega})\,, \\ \qquad \left(\nabla \times \mathbf{W}, \dfrac{1}{\sigma + \mathrm{i}\omega\varepsilon} \nabla \times \mathbf{H}\right)_{(L^2(\boldsymbol{\Omega}))^3} + \mathrm{i}\omega\mu(\mathbf{W}, \mathbf{H})_{(L^2(\boldsymbol{\Omega}))^3} = -\mathrm{i}\omega\mu(\mathbf{W}, \mathbf{M})_{(L^2(\boldsymbol{\Omega}))^3}\,. \end{cases} \tag{11}$$

2.2. 2.5D variational formulation

Herein, we focus on the case when the material properties are homogeneous along one spatial direction, e.g., y-axis. We denote the domain for this case as $\boldsymbol{\Omega} := \Omega_y \times \boldsymbol{\Omega}_{x,z}$. We perform a Fourier transform along the y-axis to

represent the 3D problem as a sequence of uncoupled 2D problems, one per Fourier mode. In this case, we define the magnetic field $\mathbf{H}$ as a series expansion using the complex exponentials:

$$\mathbf{H} \coloneqq \sum_{\beta=-\infty}^{+\infty} \mathbf{H}_\beta \exp(\mathrm{i}2\pi\beta y/L_y)\,, \tag{12}$$

where β is the Fourier mode number and $\mathbf{H}_\beta = [\mathrm{H}_x^\beta\,, \mathrm{H}_y^\beta\,, \mathrm{H}_z^\beta]$ with $\mathbf{H}_\beta : \boldsymbol{\Omega}_{x,z} \subset \mathbb{R}^2 \to \mathbb{C}^3$. Fourier modes satisfy the following orthogonality relationships, where $\delta_{i,j}$ is the Kronecker delta:

$$\frac{1}{L_y}\int_{-L_y/2}^{L_y/2} \exp(\mathrm{i}2\pi\beta_1 y/L_y)\exp(\mathrm{i}2\pi\beta_2 y/L_y)\,dy = \delta_{\beta_1,\beta_2}\,. \tag{13}$$

By employing a test function of the form

$$\mathbf{W} \coloneqq \frac{1}{L_y}\mathbf{W}_\beta \exp(\mathrm{i}2\pi\beta y/L_y)\,, \tag{14}$$

and defining the $\boldsymbol{H}(\mathrm{curl}^\beta; \boldsymbol{\Omega}_{x,z})$-conforming functional spaces

$$\boldsymbol{H}(\mathrm{curl}^\beta; \boldsymbol{\Omega}_{x,z}) \coloneqq \left\{\mathbf{W}_\beta = [\mathrm{W}_x^\beta\,, \mathrm{W}_y^\beta\,, \mathrm{W}_z^\beta] \in (L^2(\boldsymbol{\Omega}_{x,z}))^3 : \mathrm{W}_y^\beta \in H^1(\boldsymbol{\Omega}_{x,z}) \text{ and } \nabla \times [\mathrm{W}_x^\beta\,, \mathrm{W}_z^\beta] \in (L^2(\boldsymbol{\Omega}_{x,z}))^2\right\}\,, \tag{15}$$

$$\boldsymbol{H}_0(\mathrm{curl}^\beta; \boldsymbol{\Omega}_{x,z}) \coloneqq \left\{\mathbf{W}_\beta \in \boldsymbol{H}(\mathrm{curl}^\beta; \boldsymbol{\Omega}_{x,z}) : \mathbf{W}_\beta \times \mathbf{n} = \mathbf{0} \text{ on } \partial\boldsymbol{\Omega}\right\}, \tag{16}$$

we build the following variational formulation from Eq. (6) by integrating over $\boldsymbol{\Omega}_{x,z}$:

$$\begin{cases} \text{Find } \mathbf{H} = \sum_{\beta=-\infty}^{+\infty} \mathbf{H}_\beta \exp(\mathrm{i}2\pi\beta y/L_y)\,,\ \mathbf{H}_\beta \in \boldsymbol{H}_0(\mathrm{curl}^\beta; \boldsymbol{\Omega}_{x,z}) \\ \text{such that for every } \beta \in \mathbb{Z} \text{ and } \mathbf{W}_\beta \in \boldsymbol{H}_0(\mathrm{curl}^\beta; \boldsymbol{\Omega}_{x,z})\,, \\ \qquad \left(\nabla^\beta \times \mathbf{W}_\beta, \dfrac{1}{\sigma + \mathrm{i}\omega\varepsilon}\nabla^\beta \times \mathbf{H}_\beta\right)_{(L^2(\boldsymbol{\Omega}_{x,z}))^3} + \mathrm{i}\omega\mu\left(\mathbf{W}_\beta, \mathbf{H}_\beta\right)_{(L^2(\boldsymbol{\Omega}_{x,z}))^3} = -\mathrm{i}\omega\mu\left(\mathbf{W}_\beta, \mathbf{M}_\beta\right)_{(L^2(\boldsymbol{\Omega}_{x,z}))^3}\,, \end{cases} \tag{17}$$

where

$$\nabla^\beta \times \mathbf{W}_\beta \coloneqq \left[\mathrm{i}\beta\frac{2\pi}{L_y}\mathrm{W}_z^\beta - \frac{d\mathrm{W}_y^\beta}{dz}\,,\ \frac{d\mathrm{W}_x^\beta}{dz} - \frac{d\mathrm{W}_z^\beta}{dx}\,,\ \frac{d\mathrm{W}_y^\beta}{dx} - \mathrm{i}\beta\frac{2\pi}{L_y}\mathrm{W}_x^\beta\right], \tag{18}$$

and $\mathbf{M}_\beta$ is the time-harmonic magnetic source written in terms of the Fourier transform as

$$\mathbf{M}_\beta = \frac{1}{L_y}\delta(x - x_0)\delta(z - z_0)[M_x\,, M_y\,, M_z]\exp(\mathrm{i}2\pi\beta y_0/L_y)\,. \tag{19}$$

This formulation corresponds to the 2.5D variational formulation previously described by, e.g., Rodríguez-Rozas et al. (2018) and Chaumont-Frelet et al. (2018b).

Remark 1. To solve the variational problem of Eq. (17), we require an appropriate space in $\boldsymbol{\Omega}_{x,z}$ over which $\mathbf{W}_\beta$, $\nabla \times [\mathrm{W}_x^\beta\,, \mathrm{W}_z^\beta]$, and $\nabla \mathrm{W}_y^\beta$ are integrable, i.e., $[\mathrm{W}_x^\beta\,, \mathrm{W}_z^\beta] \in \boldsymbol{H}(\mathrm{curl}; \boldsymbol{\Omega}_{x,z})$ and $\mathrm{W}_y^\beta \in H^1(\boldsymbol{\Omega}_{x,z})$. Thus, we use the $\boldsymbol{H}(\mathrm{curl}^\beta; \boldsymbol{\Omega}_{x,z})$ solution space —equivalent to the $\boldsymbol{H}(\mathrm{curl}; \boldsymbol{\Omega}_{x,z}) \times H^1(\boldsymbol{\Omega}_{x,z})$ mixed space— that fulfills the mentioned requirements.

3. Borehole resistivity measurement acquisition system

We consider a logging-while-drilling (LWD) instrument equipped with transmitters (T_i) and receivers (R_j). This tool is sensitive to resistivities within the range $0.2 \sim 500\ \Omega \cdot \mathrm{m}$ (phase resistivity) and $0.2 \sim 300\ \Omega \cdot \mathrm{m}$ (amplitude resistivity) under an operating frequency between 0.1 and 2 MHz (Liu, 2017a). For the sake of simplicity, herein, we restrict to two transmitters and two receivers symmetrically located around the tool center (see Fig. 1) at an operating frequency of 2 MHz.

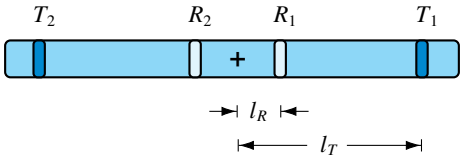


Fig. 1. A schematic LWD instrument with two transmitters and two receivers symmetrically located around the tool center.

Triaxial logging instruments generate measurements for all possible orientations of the transmitter–receiver pairs. We follow the notation presented by Davydycheva (2011) and Rodríguez-Rozas et al. (2018) to denote the magnetic field. Thus, we write $\mathrm{H}_{ZZ}^{T_i R_j} \in \mathbb{C}$ as the coaxial magnetic field in the borehole system of coordinates induced by transmitter T_i and measured at receiver R_j ($i, j = 1, 2$). We use the magnetic fields measured at R_1 and R_2 to compute the *attenuation ratio* and *phase difference*. We symmetrize the signal originating from T_1 and T_2 to obtain the *quantity of interest* Q_{ZZ} at each logging position as

$$\mathrm{Q}_{ZZ} \coloneqq \frac{1}{2}\left(\log \frac{\mathrm{H}_{ZZ}^{T_1 R_1}}{\mathrm{H}_{ZZ}^{T_1 R_2}} + \log \frac{\mathrm{H}_{ZZ}^{T_2 R_2}}{\mathrm{H}_{ZZ}^{T_2 R_1}}\right). \tag{20}$$

Then, we compute the attenuation ratio (A) and phase difference (P), respectively, as the real and imaginary parts of Q_{ZZ}:

$$A \coloneqq \mathrm{Re}(\mathrm{Q}_{ZZ}), \tag{21}$$

$$P \coloneqq \mathrm{Im}(\mathrm{Q}_{ZZ}). \tag{22}$$

We can then obtain the *apparent resistivities* based on the attenuation ratio and phase difference (ρ_A and ρ_P, respectively) using a *look-up table* algorithm. This algorithm obtains the apparent resistivities from the tool response in a homogeneous isotropic medium, which is analytically known (see Anderson, 2001).

4. Refined isogeometric analysis

In this work, we consider a *multi-field* EM problem and discretize the 2.5D variational formulation of Eq. (17) using a B-spline generalization of a curl-conforming space, introduced by Buffa et al. (2010). We first review some basic concepts of high-continuity IGA discretizations.

4.1. High-continuity IGA discretization

Given the parametric domain $\{\xi, \zeta \in \hat{\mathbf{\Omega}}_{x,z} : (0,1)^2 \subset \mathbb{R}^2\}$, we introduce the spline space $\mathcal{S}^{p_x,p_z}_{k_x,k_z}$ as

$$\mathcal{S}^{p_x,p_z}_{k_x,k_z} \coloneqq \text{span}\left\{B^{p_x,p_z}_{i,j}\right\}^{n_x-1,n_z-1}_{i=0,j=0} \,, \tag{23}$$

where n, p, and k with their indices are the number of degrees of freedom, polynomial degree, and continuity of basis functions in x and z directions, respectively, resulting in $n_e \coloneqq n - p$ number of elements in each direction. The bivariate basis functions are

$$B^{p_x,p_z}_{i,j} \coloneqq B^{p_x}_i(\xi) \otimes B^{p_z}_j(\zeta) \,, \qquad i = 0,1,...,n_x - 1 \,, \quad j = 0,1,...,n_z - 1 \,, \tag{24}$$

where the univariate bases are expressed by the Cox–De Boor recursion formula (Piegl and Tiller, 1997) as

$$B^0_i(\xi) = \begin{cases} 1 \,, & \xi_i \le \xi < \xi_{i+1} \,, \\ 0 \,, & \text{otherwise} \,, \end{cases} \tag{25}$$

$$B^p_i(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} B^{p-1}_i(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} B^{p-1}_{i+1}(\xi) \,, \tag{26}$$

and spanned over the respective knot sequences in x and z directions, given by

$$\Xi = [\underbrace{0,0,...,0}_{p_x+1}, \xi_{p_x+1}, \xi_{p_x+2}..., \xi_{n_x-1}, \underbrace{1,1,...,1}_{p_x+1}] \,, \tag{27}$$

$$\mathrm{Z} = [\underbrace{0,0,...,0}_{p_z+1}, \zeta_{p_z+1}, \zeta_{p_z+2}..., \zeta_{n_z-1}, \underbrace{1,1,...,1}_{p_z+1}] \,. \tag{28}$$

We assume single multiplicities for all knots, providing maximum continuity $k = p - 1$ for the IGA discretization.

Fig. 2 illustrates the $\boldsymbol{H}(\text{curl}) \times H^1$ IGA discrete space in $\mathbf{\Omega}_{x,z}$ along with the univariate basis functions of the respective vector fields. For brevity, herein and in the following, we exclude the superscript β in referring to the components of the magnetic field.

We define the spaces in the parametric domain and introduce the appropriate transformations to obtain the discretization on the physical domain. We start with the set of discrete spaces in the parametric domain, given by

$$\hat{\mathcal{V}}^{\text{curl}}_h(\hat{\mathbf{\Omega}}_{x,z}) \coloneqq \mathcal{S}^{p-1,p}_{k-1,k} \times \mathcal{S}^{p,p-1}_{k,k-1} \,, \tag{29}$$

$$\hat{\mathcal{Q}}^{\text{grad}}_h(\hat{\mathbf{\Omega}}_{x,z}) \coloneqq \mathcal{S}^{p,p}_{k,k} \,. \tag{30}$$

By defining $\mathbf{F} : \hat{\mathbf{\Omega}}_{x,z} \to \mathbf{\Omega}_{x,z}$ as the geometric mapping from the parametric domain onto the physical domain, and $D\mathbf{F}$ as its Jacobian, we introduce the set of discrete spaces in the physical domain:

$$\mathcal{V}^{\text{curl}}_h(\mathbf{\Omega}_{x,z}) \coloneqq \left\{\mathbf{H}_{x,z} = [\mathrm{H}_x \,, \mathrm{H}_z] \in \boldsymbol{H}(\text{curl}; \mathbf{\Omega}_{x,z}) \cap \boldsymbol{H}^1(\mathbf{\Omega}_{x,z}) : \iota^{\text{curl}}(\mathbf{H}_{x,z}) = \hat{\mathbf{H}}_{x,z} \in \hat{\mathcal{V}}^{\text{curl}}_h(\hat{\mathbf{\Omega}}_{x,z})\right\} \,, \tag{31}$$

$$\mathcal{Q}^{\text{grad}}_h(\mathbf{\Omega}_{x,z}) \coloneqq \left\{\mathrm{H}_y \in H^1(\mathbf{\Omega}_{x,z}) : \iota^{\text{grad}}(\mathrm{H}_y) = \hat{\mathrm{H}}_y \in \hat{\mathcal{Q}}^{\text{grad}}_h(\hat{\mathbf{\Omega}}_{x,z})\right\} \,, \tag{32}$$

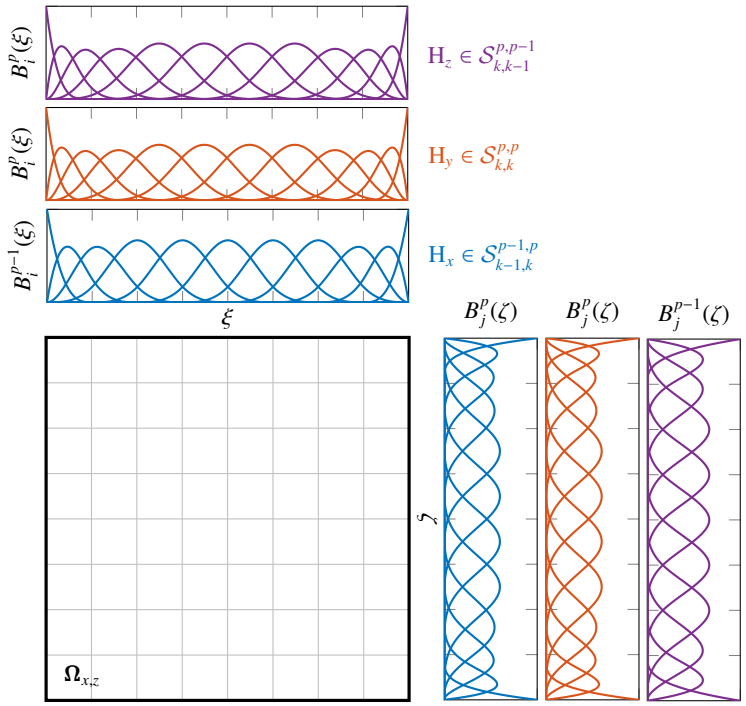


Fig. 2. Example of the $\boldsymbol{H}(\mathrm{curl}) \times H^1$ space for a 2.5D formulation discretized by C^{p-1} IGA with uniform 8×8 elements in $\boldsymbol{\Omega}_{x,z}$, polynomial degree $p = 4$, and continuity $k = 3$. The univariate basis functions of H_x, H_y, and H_z are shown in blue, red, and purple, respectively. Thin gray lines in the mesh skeleton denote the high-continuity element interfaces.

where we use the following curl- and grad-preserving pullback mappings (Garcia et al., 2019; Buffa et al., 2010):

$$\iota^{\mathrm{curl}}(\mathbf{H}_{x,z}) \coloneqq (D\mathbf{F})^T(\mathbf{H}_{x,z} \circ \mathbf{F}), \tag{33}$$

$$\iota^{\mathrm{grad}}(\mathrm{H}_y) \coloneqq \mathrm{H}_y \circ \mathbf{F}. \tag{34}$$

Thus, by defining the discrete space

$$\mathcal{H}_{h,0}(\boldsymbol{\Omega}_{x,z}) \coloneqq \left\{ \mathbf{H}_{\beta,h} \in \mathcal{V}_h^{\mathrm{curl}}(\boldsymbol{\Omega}_{x,z}) \times \mathcal{Q}_h^{\mathrm{grad}}(\boldsymbol{\Omega}_{x,z}) : \mathbf{H}_{\beta,h} \times \mathbf{n} = \mathbf{0} \text{ on } \partial\boldsymbol{\Omega} \right\}, \tag{35}$$

we write the discrete form of Eq. (17) as follows (subscript h refers to discrete solution):

$$\begin{cases} \text{Find } \mathbf{H}_h = \sum_{\beta=-\infty}^{+\infty} \mathbf{H}_{\beta,h} \exp(\mathrm{i}2\pi\beta y/L_y), \ \mathbf{H}_{\beta,h} \in \mathcal{H}_{h,0}(\boldsymbol{\Omega}_{x,z}) \\ \text{such that for every } \beta \in \mathbb{Z} \text{ and } \mathbf{W}_{\beta,h} \in \mathcal{H}_{h,0}(\boldsymbol{\Omega}_{x,z}), \\ \qquad \left(\nabla^\beta \times \mathbf{W}_{\beta,h}, \frac{1}{\sigma + \mathrm{i}\omega\varepsilon} \nabla^\beta \times \mathbf{H}_{\beta,h} \right)_{(L^2(\boldsymbol{\Omega}_{x,z}))^3} + \mathrm{i}\omega\mu \left(\mathbf{W}_{\beta,h}, \mathbf{H}_{\beta,h} \right)_{(L^2(\boldsymbol{\Omega}_{x,z}))^3} = -\mathrm{i}\omega\mu \left(\mathbf{W}_{\beta,h}, \mathbf{M}_\beta \right)_{(L^2(\boldsymbol{\Omega}_{x,z}))^3}. \end{cases} \tag{36}$$

4.2. rIGA discretization

The refined isogeometric analysis (rIGA) is a discretization technique that optimizes the performance of direct solvers. In particular, rIGA preserves the optimal convergence order of the direct solvers with respect to a fixed

number of elements in the domain. Garcia et al. (2017) first presented this strategy for H^1 spaces and then extended it to $\boldsymbol{H}(\text{curl})$, $\boldsymbol{H}(\text{div})$, and L^2 spaces (see Garcia et al., 2019). Starting from the high-continuity C^{p-1} IGA discretization, rIGA reduces the continuity of certain basis functions by increasing the multiplicity of the respective existing knots. Hence, the computational domain is subdivided into high-continuity macroelements interconnected by low-continuity hyperplanes. These hyperplanes coincide with the locations of the *separators* at different partitioning levels of the multifrontal direct solvers. Thus, rIGA reduces the computational cost of matrix factorization when solving PDE systems in comparison to IGA and FEA.

For multi-field problems discretized using $\boldsymbol{H}(\text{curl};\boldsymbol{\Omega}_{x,z}) \times H^1(\boldsymbol{\Omega}_{x,z})$ spaces, we preserve the commutativity of the *de Rham* diagram (Demkowicz et al., 2000) by reducing the continuity in $k-1$ degrees. To achieve this, we use both C^0 and C^1 hyperplanes and reduce the continuity across the interface between the subdomains (i.e., macroelements). Fig. 3 depicts the rIGA discretization of the $\boldsymbol{H}(\text{curl};\boldsymbol{\Omega}_{x,z}) \times H^1(\boldsymbol{\Omega}_{x,z})$ space of Fig. 2 after one level of symmetric partitioning, which results in macroelements containing 4×4 elements.

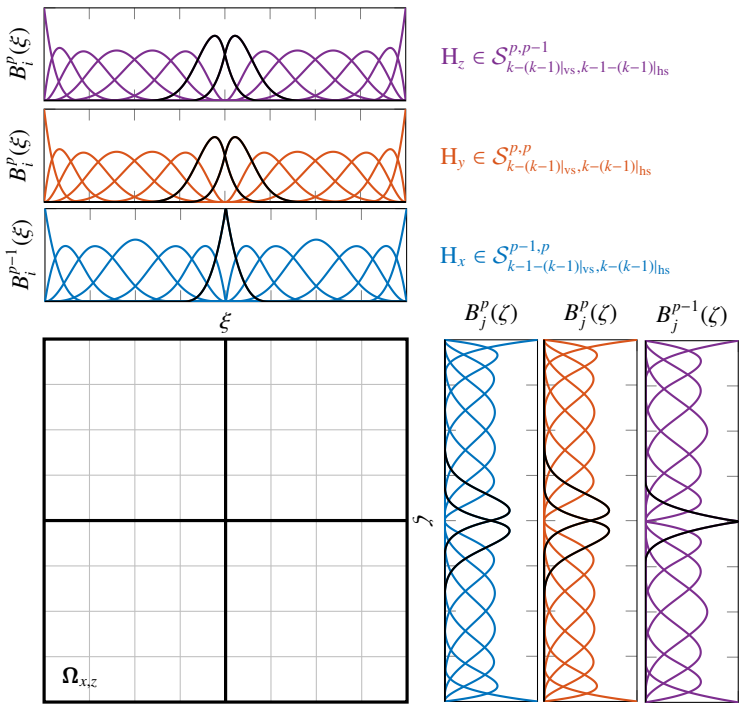


Fig. 3. $\boldsymbol{H}(\text{curl}) \times H^1$ rIGA space in $\boldsymbol{\Omega}_{x,z}$, associated with the 8×8 domain of Fig. 2 with $p = 4$ and $k = 3$, after one level of symmetric partitioning by the rIGA discretization that results in 4×4 macroelements. rIGA reduces the continuity of basis functions by $k-1$ degrees across the macroelement separators (the low-continuity bases are shown in black). Thin gray lines in the mesh skeleton denote the high-continuity element interfaces, while thick black lines illustrate the macroelement boundaries. We refer to the vertical and horizontal separators as "vs" and "hs", respectively.

Previous works show rIGA discretizations provide significant improvements in the solution time and memory requirements. In particular, the rIGA solution is obtained up to $\mathcal{O}(p^2)$ faster in large domains —and $\mathcal{O}(p)$ faster in small domains— than the IGA solution. In comparison to traditional FEA with the same number of elements, rIGA

provides even larger improvements. rIGA also reduces the memory requirements since the rIGA LU factors have fewer nonzero entries than the IGA LU factors. Finally, rIGA improves the approximation error with respect to IGA since the continuity reduction of basis functions enriches the Galerkin space (see Garcia et al., 2017, 2019; Hashemian et al., 2021).

5. Implementation details

We implement discrete $\boldsymbol{H}(\text{curl};\boldsymbol{\Omega}_{x,z}) \times H^1(\boldsymbol{\Omega}_{x,z})$ spaces using PetIGA-MF (Sarmiento et al., 2017), a multi-field extension of PetIGA (Dalcin et al., 2016), a high-performance isogeometric analysis implementation based on PETSc (portable extensible toolkit for scientific computation) (Balay et al., 1997). PetIGA-MF allows the use of different spaces for each field of interest and employs data management libraries to condense the data of multiple fields in a single object, thus simplifying the discretization construction. This framework also allows us to investigate both IGA and rIGA discretizations in our 2.5D problem with different numbers of elements, different polynomial degrees of the B-spline spaces, and different partitioning levels of the mesh.

We use Intel MKL with PARDISO (Petra et al., 2014a,b) as our sparse direct solver package to construct LU factors for solving the linear systems of equations. PARDISO employs supernode techniques to perform the matrix factorization (see, e.g., Schenk et al., 2000; Schenk and Gärtner, 2004). It provides parallel factorization using OpenMP directives (Dagum and Menon, 1998) and uses the automatic matrix reordering provided by METIS (Karypis and Kumar, 1998). We executed all tests on a workstation equipped with two Intel Xeon Gold 6230 CPUs at 2.10 GHz with 40 threads per CPU.

We employ a tensor-product mesh with variable element sizes (see Fig. 4). At each logging position, the computational mesh has a fine subgrid in the central part of the domain with element size equal to $h \times h$. This subgrid is surrounded by another tensor-product grid whose element sizes grow slowly until reaching the boundary. Let n_e be the number of elements in each direction and $n_c < n_e$ is the number of elements in each direction located at the central part of the domain. We use the power function of Eq. (37) to model the geometrical progression of the mesh. We follow the algorithm presented by Rodríguez-Rozas et al. (2018) to find suitable values for n_c and growth rate r. Starting from the center point of a symmetric domain, we obtain the size of the i-th element in each direction h_i ($i = 1, 2, ..., n_e/2$) as follows:

$$h_i = \begin{cases} h\,, & 1 \le i \le n_c/2\,, \\ hr^{(i-n_c/2)}\,, & n_c/2 < i \le n_e/2\,. \end{cases} \tag{37}$$

Remark 2. For each logging position, we perform a single *symbolic* factorization common to all Fourier modes, followed by a numerical factorization per Fourier mode. Once we solve the system of equations for the first transmitter, we update the right-hand side of Eq. (36) to solve for the magnetic field induced by the second transmitter and use backward substitution. Hence, we perform only one LU factorization per Fourier mode per logging position for both transmitters.

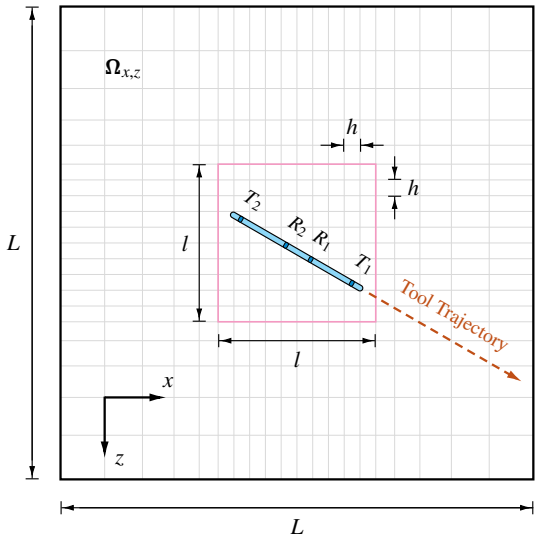


Fig. 4. A drawing of the computational domain $\Omega_{x,z}$ and the tool trajectory. The central subgrid bounded by a magenta box is composed of a set of fine elements located in the proximity of the logging instrument. The remaining elements grow smoothly in size until reaching the boundary.

Remark 3. The convergence of the Fourier series leads to a fast decay of the real and imaginary parts of H_{ZZ} for higher Fourier modes (see Rodríguez-Rozas and Pardo, 2016). Thus, we truncate the series of Eq. (12) when the magnetic field at the receivers is sufficiently small, such that $\beta \in [-N_f, N_f]$, being $2N_f + 1$ the total number of Fourier modes. Due to the symmetry of the media along the y direction, we only consider $\beta \in [0, N_f]$.

6. Numerical Results

In this section, we first assess the accuracy of the rIGA approach in a homogeneous medium. We also investigate the computational efficiency of the rIGA framework in comparison with IGA and FEA approaches. Then, we consider two model problems consisting of high-angle wells crossing spatially heterogeneous media with multiple geological faults. Finally, we produce our synthetic training dataset as a preliminary stage for DL inversion (see Appendix A for more details regarding the numerical codes and required packages). In our simulations, we consider one operational mode of a commercial logging tool (Zhou, 2016) with $l_R = 10.16$ cm and $l_T = 56.8325$ cm (see Fig. 1). We select the free space electric permittivity and magnetic permeability as $\varepsilon = 8.85 \times 10^{-12}$ F $\cdot$ m^{-1} and $\mu = 4\pi \times 10^{-7}$ N $\cdot$ A^{-2}, respectively. We also consider a transmitter frequency $f = 2$ MHz.

6.1. Homogeneous medium

We assume the logging instrument is placed in a homogeneous medium with resistivity $\rho = 1/\sigma = 100\ \Omega \cdot$ m. This high-resistivity case is numerically more challenging than low-resistivity cases since it requires a larger number of Fourier modes and numerical precision. We consider a cube domain of length $L = 18$ m for our first case study.

6.1.1. *Accuracy assessment*

To assess the accuracy and select certain discretization parameters, we compare the numerical attenuation ratio, A, and phase difference, P, given by Eqs. (21) and (22), with the expected (i.e., exact) values, A_e and P_e, obtained from $\rho_e = 1/\sigma$. Fig. 5 shows the numerical errors, i.e., $|1 - A/A_e|$ and $|1 - P/P_e|$, as a function of the number of Fourier modes when computing attenuation ratio and phase difference in a homogeneous medium. Herein, we select a domain with 64×64 elements to ensure a fast numerical solution for our measurements. We compare the results of the high-continuity C^{p-1} IGA with FEA and also with an rIGA discretization that employs 8×8 macroelements. This macroelement size provides the fastest results for moderate size domains (see Garcia et al., 2017). We consider three different mesh sizes —$h = 0.025$ m, 0.033 m, and 0.050 m— and different polynomial degrees —$p = 3$, 4, and 5. The best results correspond to $h = 0.025$ m (blue lines in the figure). We also observe that rIGA and FEA discretizations deliver lower errors compared to their IGA counterparts —when the same number of elements and polynomial degree are considered— taking into account that rIGA provides solutions with higher computational efficiency (see Section 6.1.2).

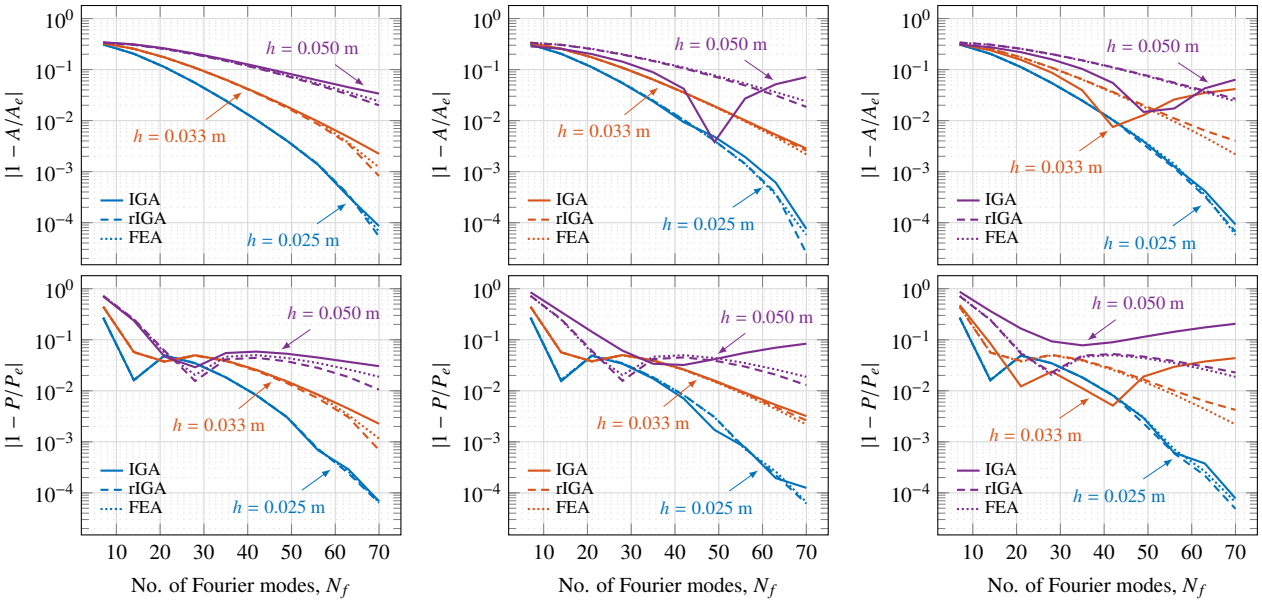


Fig. 5. Numerical errors when computing attenuation ratio, A, and phase difference, P, in a homogeneous medium using IGA and rIGA discretizations, obtained by for a 64×64 mesh with different element sizes h and polynomial degrees p.

To investigate the decay of the solution for each Fourier mode, we compare numerical results with the analytical 2.5D solution in the homogeneous medium presented by Rodríguez-Rozas et al. (2018). In particular, given M_z as the only nonzero component of the magnetic source, it is possible to analytically determine the coaxial magnetic field for each Fourier mode as follows:

$$\mathrm{H}_{ZZ}(\beta) = -\mathrm{i}\omega\mu\sigma\tau_{\beta_z} + \frac{\partial^2 \tau_{\beta_z}}{\partial z^2}\,, \tag{38}$$

with

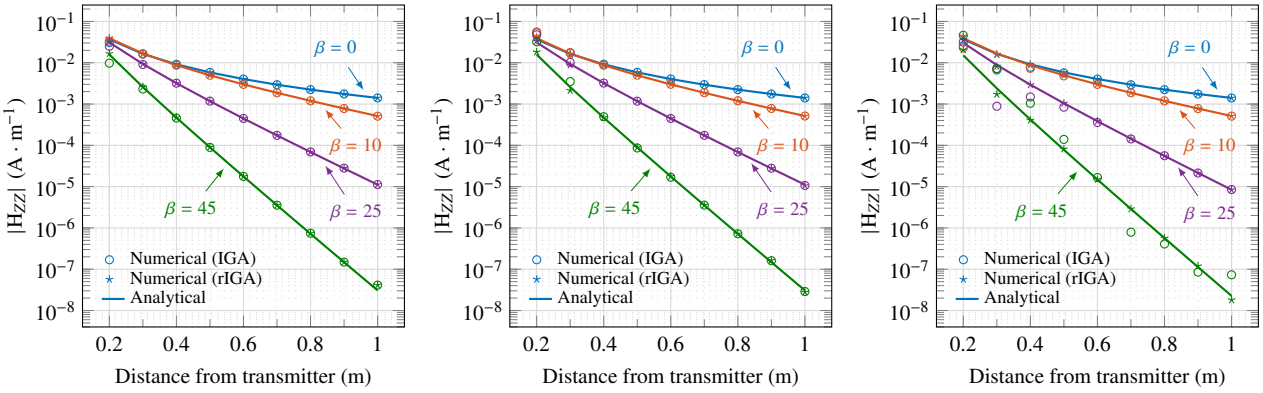
$$\tau_{\beta_z} = \frac{M_z}{2\pi}\frac{1}{L_y} K_0(\mathrm{C}R)\exp(\mathrm{i}2\pi\beta y_0/L_y)\,, \tag{39}$$

where $K_0(\cdot)$ is the modified Bessel function of the second kind of order zero, and

$$\mathrm{C} = \left(2\pi\beta/L_y\right)^2 + \mathrm{i}\omega\mu\sigma\,, \tag{40}$$

$$R = \sqrt{(x-x_0)^2 + (z-z_0)^2}\,. \tag{41}$$

Fig. 6 compares the decay of the numerical coaxial magnetic field $\mathrm{H}_{ZZ}(\beta)$ with its analytical counterpart for some Fourier modes. Using a domain with 64×64 elements and $h = 0.025$ m, we monitor the decay of the propagated waves at distances within the interval $[0.2, 1.0]$ m from the transmitters to ensure that the solutions at both receivers properly approximate the analytical ones. Results show that rIGA discretizations deliver increased accuracy for all tested polynomial degrees.



(a) $p = 3$ (b) $p = 4$ (c) $p = 5$

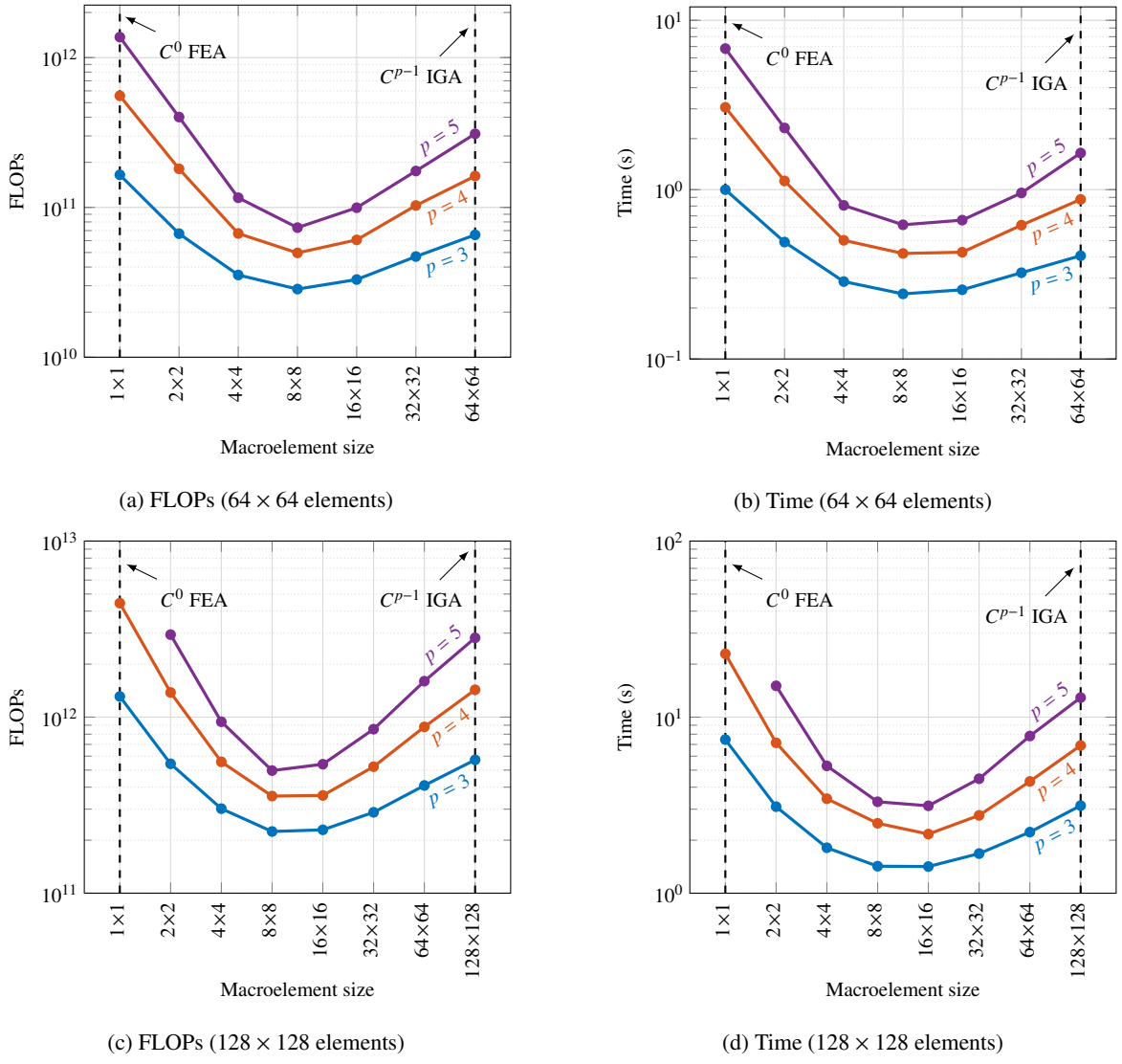
Fig. 6. Comparison of the decay of the numerical and analytical coaxial magnetic fields for some Fourier modes, obtained in a grid of 64×64 elements with $h = 0.025$ m and different polynomial degrees.

6.1.2. Computational efficiency

Garcia et al. (2017, 2019) provide theoretical cost estimates of solving H^1 and $\boldsymbol{H}(\mathrm{curl})$ discrete spaces, respectively. Herein, we add these estimates to predict the cost of discretizing the $\boldsymbol{H}(\mathrm{curl}; \boldsymbol{\Omega}_{x,z}) \times H^1(\boldsymbol{\Omega}_{x,z})$ space appearing in our 2.5D EM problem. We conclude that the cost of LU factorization of the rIGA matrix for this combined space is between $\mathcal{O}(p)$ and $\mathcal{O}(p^2)$ times smaller than that for IGA. Details are omitted for the sake of simplicity.

To numerically assess the computational efficiency confirming the aforementioned theoretical results, we consider two different grids in $\boldsymbol{\Omega}_{x,z}$ with 64×64 and 128×128 elements, respectively. Using continuity reduction, we split

the mesh symmetrically into macroelements whose sizes are powers of two. In this context, the maximum-continuity C^{p-1} IGA discretization is composed of one macroelement containing the entire grid, while C^0 FEA with minimum continuity across all element interfaces is composed of macroelements that contain only one element. Fig. 7 shows the number of FLOPs and time required to solve the borehole resistivity problem for each Fourier mode per logging position. We compare the computational costs for different polynomial degrees and different continuity reduction levels of basis functions. The cost of rIGA reaches the minimum with 8×8 macroelements almost in all cases, confirming the theoretical estimates obtained from the results of Garcia et al. (2017).



(a) FLOPs (64×64 elements)

(b) Time (64×64 elements)

(c) FLOPs (128×128 elements)

(d) Time (128×128 elements)

Fig. 7. Computational cost in terms of FLOPs and time for solving a 2.5D borehole resistivity problem per logging position per Fourier mode. We test rIGA discretizations with two different grids of 64×64 and 128×128 elements. The computational times correspond to the use of parallel solver PARDISO using two threads.

Our numerical tests show that for a moderate size 2.5D problem, the reduction in the number of FLOPs is $\mathcal{O}(p)$ with respect to IGA. When compared to FEA, rIGA delivers larger improvement factors. These improvement factors in terms of FLOPs also hold in terms of time when performing a sequential factorization. In our parallel PARDISO solver, we observe a small degradation of the rIGA improvement factors in terms of times in comparison to those obtained in terms of FLOPs (see Table 1).

Table 1. Computational cost for the 2.5D borehole resistivity measurements per logging position per Fourier mode. We report the solution time and FLOPs when using C^{p-1} IGA, rIGA with 8×8 macroelements, and C^0 FEA with the same number of elements and polynomial degree. The computational times correspond to the use of parallel solver PARDISO using two threads.

Domain size	Polynomial degree	Discretization method	Number of FLOPs	Improvement factor (FLOPs)		Time (s)	Improvement factor (time)	
		IGA	6.56e+10			0.407		
	3	rIGA	2.85e+10	IGA/rIGA	2.30	0.242	IGA/rIGA	1.68
		FEA	1.65e+11	FEA/rIGA	5.79	0.999	FEA/rIGA	4.12
		IGA	1.62e+11			0.875		
64×64	4	rIGA	4.97e+10	IGA/rIGA	3.26	0.419	IGA/rIGA	2.09
		FEA	5.56e+11	FEA/rIGA	11.19	3.061	FEA/rIGA	7.29
		IGA	3.10e+11			1.645		
	5	rGA	7.33e+10	IGA/rIGA	4.23	0.620	IGA/rIGA	2.65
		FEA	1.37e+12	FEA/rIGA	18.69	6.806	FEA/rIGA	10.98
		IGA	5.72e+11			3.144		
	3	rIGA	2.24e+11	IGA/rIGA	2.55	1.423	IGA/rIGA	2.21
		FEA	1.31e+12	FEA/rIGA	5.85	7.456	FEA/rIGA	5.24
		IGA	1.43e+12			6.903		
128×128	4	rIGA	3.56e+11	IGA/rIGA	4.02	2.495	IGA/rIGA	2.77
		FEA	4.44e+12	FEA/rIGA	12.47	22.885	FEA/rIGA	9.17
		IGA	2.82e+12			12.911		
	5	rGA	4.97e+11	IGA/rIGA	5.67	3.305	IGA/rIGA	3.91
		FEA	–	FEA/rIGA	–	–	FEA/rIGA	–

6.2. Heterogeneous media

We further examine the accuracy of our rIGA approximation over two synthetic heterogeneous model problems.

6.2.1. One geological fault

We consider the model problem of Fig. 8 with a constant dip angle of 80°. We consider the LWD instrument described in Section 3 and simulate measurements recorded over 200 equally-spaced logging positions throughout the well trajectory.

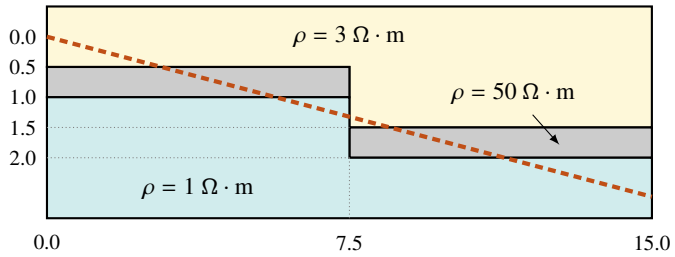


Fig. 8. Model problem with a constant dip angle of 80° passing through a geological fault and three different materials (well trajectory is highlighted by a red dashed line). Dimensions are in meters.

Fig. 9 shows the apparent resistivities based on the attenuation ratio and phase difference (ρ_A and ρ_P, respectively). We obtain the results using $N_f = 70$ and an rIGA discretization with 64×64 elements, $p = 4$, and 8×8 macroelements. Results are in good agreement with those presented by Rodríguez-Rozas et al. (2018).

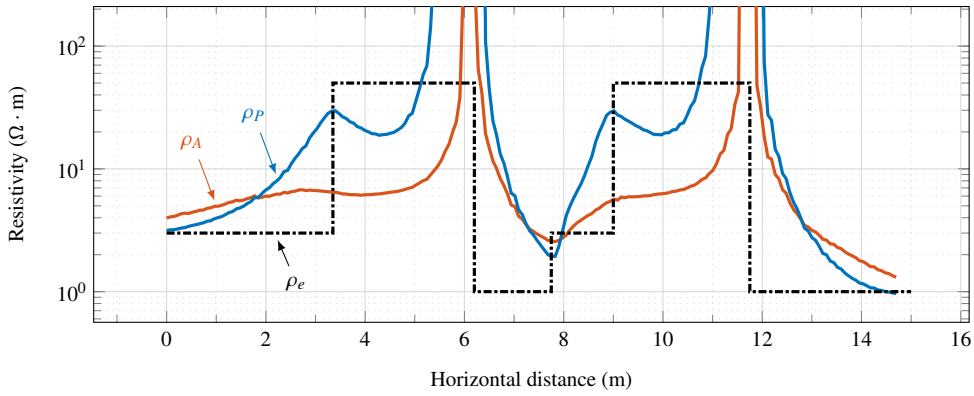


Fig. 9. Apparent resistivities based on the attenuation ratio, ρ_A, and phase difference, ρ_P, for the first model problem, compared with the real resistivity, ρ_e. We obtain the results using an rIGA discretization with 64×64 elements, $p = 4$, and 8×8 macroelements.

6.2.2. Two geological faults and inclined layers

Fig. 10 shows the second model problem containing two geological faults and inclined layers. The logging trajectory starts from a sandstone layer with a resistivity of $\rho = 3\ \Omega \cdot \mathrm{m}$, and passes through an oil-saturated layer with $\rho = 100\ \Omega \cdot \mathrm{m}$. The tool trajectory also passes through a water-saturated layer with $\rho = 0.5\ \Omega \cdot \mathrm{m}$.

In particular, inclined layers produce the so-called *staircase* approximations (Cangellaris and Wright, 1991). This phenomenon occurs because the physical interfaces of the conductivity model are not aligned with the element edges. Thus, the conductivity parameter takes different values inside some elements of the mesh. To tackle this issue, discretization techniques using nonfitting grids (Chaumont-Frelet et al., 2018a,b) are available, but they have not been considered here for simplicity.

Fig. 11 shows the apparent resistivities based on the attenuation ratio and phase difference throughout the logging

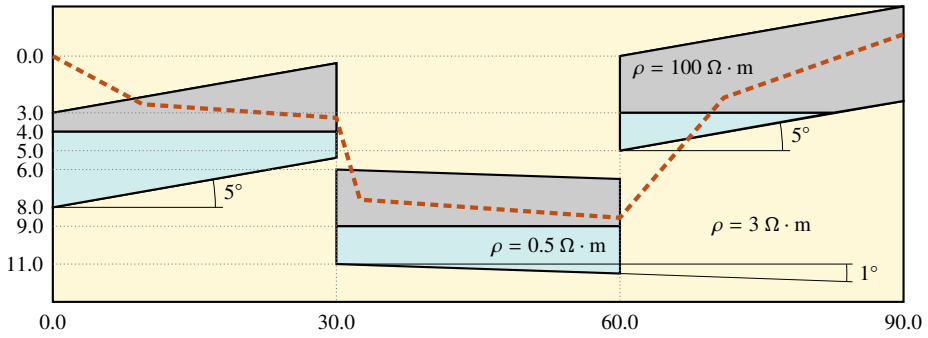


Fig. 10. Second model problem with two geological faults and inclined layers. The tool trajectory (red dashed line) has different dip angles and passes through sandstone (yellow), oil-saturated (gray) and water-saturated (green) layers. Dimensions are in meters.

trajectory and compares their value with the exact resistivity. We simulate the resistivities at 1,080 logging positions with $N_f = 70$. We use an rIGA discretization with 64×64 elements, $p = 4$, and 8×8 macroelements.

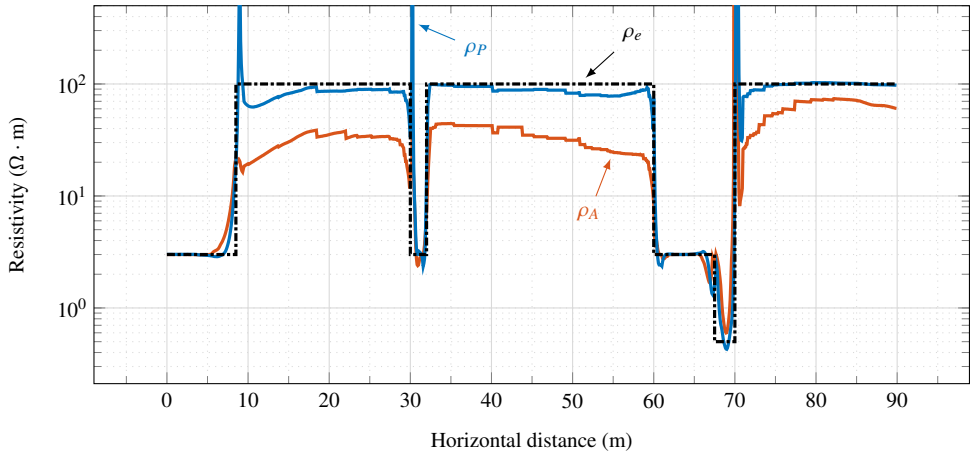


Fig. 11. Apparent resistivities based on the attenuation ratio, ρ_A, and phase difference, ρ_P, for the second model problem, compared with the real resistivity, ρ_e. We obtain the results using an rIGA discretization with 64×64 elements, $p = 4$, and 8×8 macroelements.

6.3. Database generation for DL inversion

To produce our synthetic training dataset for DL inversion, we consider heterogeneous medium containing three different layers and six varying parameters at each logging position, as described in Fig. 12 and Table 2. We select three different electrical conductivities: σ_c for the central layer, and σ_u and σ_l for the upper and lower layers, respectively. We assume the tool center is always within the middle layer and has vertical distances of d_u and d_l from the upper and lower layers, respectively. The sixth varying parameter is the dip angle, φ, measured from the vertical direction.

In here, we create a dataset of 100,000 samples and compute the apparent resistivities obtained from random com-

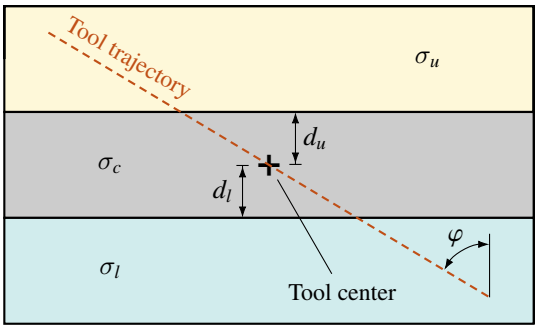


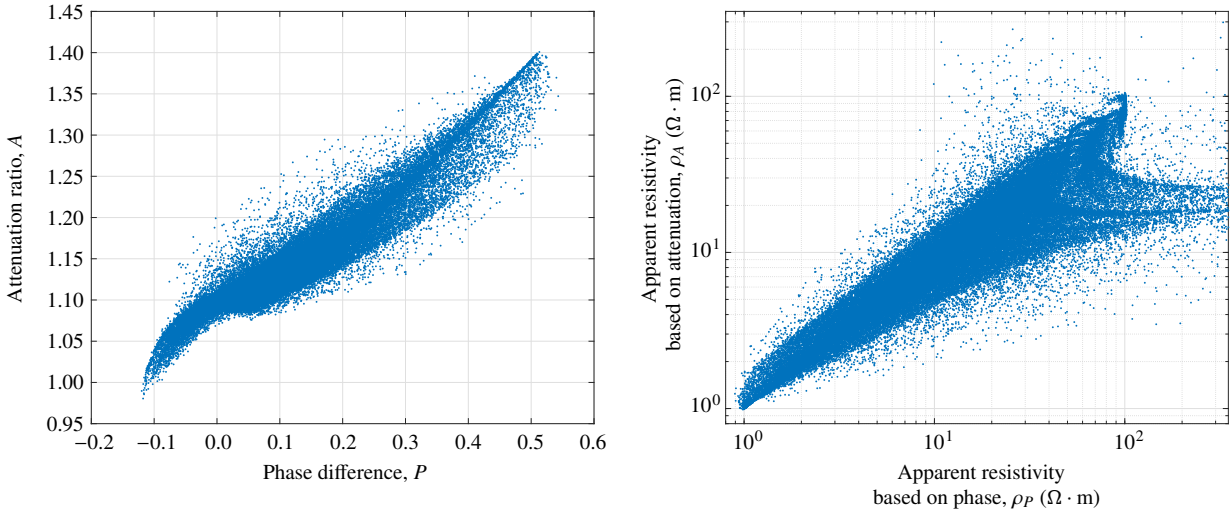
Fig. 12. Varying parameters at each logging position when producing the training dataset for DL inversion.

Table 2. Varying parameters employed to generate the training dataset for DL inversion.

Varying parameters		Interval
Electrical conductivity of the central layer	$\log_{10}(\sigma_c)$	$[-2, 0]$
Electrical conductivity of the upper layer	$\log_{10}(\sigma_u)$	$[-2, 0]$
Electrical conductivity of the lower layer	$\log_{10}(\sigma_l)$	$[-2, 0]$
Distance of the tool center from the upper layer	$\log_{10}(d_u)$	$[-2, 1]$
Distance of the tool center from the lower layer	$\log_{10}(d_l)$	$[-2, 1]$
Dip angle between the tool and the layered media	φ	$[80°, 100°]$

binations within a given range of resistivities $\rho = 1/\sigma \in [1, 100]\ \Omega \cdot \text{m}$ (see Table 2). For generating the dataset, we use two different types of parallelization. One parallelization is related to the parallel factorization of the direct solver, and the other is the trivial parallelization based on scheduling the solutions of independent Earth models onto different processors. Using 40 threads, we solve for 20 different Earth models, each executing over two threads. Table 1 shows that the required time for matrix factorization of the 2.5D EM problem using optimal rIGA discretization with 64×64 grid, $p = 4$, and 8×8 macroelements is about 0.42 seconds per Fourier mode. Considering $N_f = 70$, and the additional time required for pre/postprocessing and inter-thread communications, each set of independent runs (consists of 20 different Earth models) takes about 40 seconds. Thus, we perform 5,000 sequential runs to construct our 100,000 samples in about 56 hours. To create a larger database, we could execute over a cluster of hundreds of CPUs/threads, expecting a perfect parallel scalability.

Fig. 13a depicts the graphs of attenuation ratio, A, versus phase difference, P, obtained from the 100,000 Earth models when using rIGA discretization for generating the database. Since there is a strong correlation between A and P, the data distribution on the plot follows an almost straight line. We also display in Fig. 13b the correlation between apparent resistivities based on attenuation ratio and phase difference.



(a) (b)

Fig. 13. (a) Attenuation ratio vs. phase difference, and (b) apparent resistivity based on attenuation vs. apparent resistivity based on phase, obtained for the 100,000 Earth models. We use rIGA discretization with 64×64 elements, $p = 4$, and 8×8 macroelements for generating the database.

7. Discussions

Herein, we discuss some of the topics observed during the course of this research:

FEA vs. IGA and rIGA. When using FEA, the solution space is characterized by basis functions that have support over up to two elements in each spatial direction. Considering n_e as the number of elements in each direction, p as the polynomial degree of basis functions, and d as the space dimension, we deal with a linear system of equations that has $\mathcal{O}((n_e p)^d)$ unknowns. Whereas in IGA, each basis function is spanned over $p + 1$ elements. As a result, not only we have smoother solution space using IGA (because of higher continuity of basis functions), but also the system of equations has lower number of unknowns, namely, $\mathcal{O}((n_e + p)^d)$, resulting in cheaper computations compared to FEA. In practice, if we want to reduce the FEA computational cost, we may decrease either the degree p or the number of elements n_e. Either of these deteriorates the solution accuracy. Herein, we propose to use the "refined IGA" (rIGA). Thus, we conserve the desirable properties of IGA while reducing the solution cost of direct solvers by decreasing the interconnection of elements in system matrices.

Different types of parallelization. One may consider different types of parallelization, e.g., along the Fourier modes, parallel factorization using parallel direct solvers, and the trivial parallelization based on independent Earth models. In this work, we consider the latter two: parallelizations related to the parallel factorization, and scheduling the solutions of independent Earth models onto different processors. Since each new Earth model entails a new symbolic factorization, the advantage of using these parallelizations lies in the fact that we can use one symbolic factorization for all Fourier modes associated with the same processor.

Synthetic vs. real dataset. We employ synthetic data because we lack from massive annotated real data, i.e., inverted Earth models. Real data is typically sparse (i.e., insufficient) to train a neural network for DL inversion. Nonetheless, we may complement the synthetic data with the real data to benefit from both. From the computational point of view, borehole measurements satisfy Maxwell's equations. However, due to instrumentation noise and geometrical complexity, we may be unable to use real data in some occasions and, instead, we employ synthetic data to reproduce real data. It is usual in the industry to employ synthetic data for these purposes and also to add noise to the synthetic measurements. In principle, it is possible to use a transfer learning algorithm to adapt the synthetic data to the real data and then, compensate for the noise using a few real samples and a pretrained model based on the synthetic data.

2D Earth models for database generation. Using more detailed 2D Earth models do not affect the computational costs, but may affect the solution accuracy if we deal with geological faults that are not aligned with the computational grid. Thus, we need to align them or use some kind of numerical strategy, e.g., by using nonfitting grids (Chaumont-Frelet et al., 2018a,b), in order to improve the accuracy.

8. Conclusions

We propose the use of refined isogeometric analysis (rIGA) discretizations for generating a massive synthetic database for deep learning inversion of 2.5D borehole electromagnetic measurements. Such a large database is essential for layer-by-layer estimation of the inverted Earth models, which may be used for real-time adjustments of the well trajectory during geosteering operations.

rIGA delivers computational savings of up to $\mathcal{O}(p)$ compared to the high-continuity isogeometric analysis (IGA). When compared to a traditional finite element analysis (FEA) with the same mesh size and polynomial degree, rIGA provides higher improvement factors. At the same time, rIGA provides sufficiently accurate solutions for geosteering purposes.

To create a dataset for deep learning inversion, we first selected certain discretization parameters based on the results of several homogeneous solutions. Then, we checked the accuracy over homogeneous and heterogeneous media. Finally, we generated a *meaningful* synthetic database composed of 100,000 Earth models with the corresponding measurements in about 56 hours using a workstation equipped with two CPUs. Herein, by meaningful synthetic database we mean a set of borehole measurements that covers adequate ranges of variation of the involved variables (i.e., Earth parameterization models) and should be sufficient for a robust DL inversion.

As future work, we propose the use of artificial intelligence based techniques to perform the inversion of a large set of borehole resistivity measurements with 2D Earth models containing multiple geological faults. The required dataset for such inversion can be created synthetically and complemented by some real Earth models. To generate the synthetic database, we propose to execute the presented method over a cluster of hundreds of CPUs/threads. We will also employ more complex Earth model parameterizations including anisotropic layers.

Acknowledgment

This work has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 777778 (MATHROCKS), the European POCTEFA 2014-2020 Project PIXIL (EFA362/19) by the European Regional Development Fund (ERDF) through the Interreg V-A Spain-France-Andorra programme, the Project of the Spanish Ministry of Science and Innovation with reference PID2019-108111RB-I00 (FEDER/AEI), the BCAM "Severo Ochoa" accreditation of excellence (SEV-2017-0718), and the Basque Government through the BERC 2018-2021 programme, the two Elkartek projects 3KIA (KK-2020/00049) and MATHEO (KK-2019-00085), the grant "Artificial Intelligence in BCAM number EXP. 2019/00432", and the Consolidated Research Group MATHMODE (IT1294-19) given by the Department of Education. The authors also acknowledge the computer resources and the technical support provided by Barcelona Supercomputing Center through the "MareNostrum4" cluster and under the activity ID "IM-2020-3-0009".

References

Abubakar, A., Habashy, T., Druskin, V., Alumbaugh, D., Zerelli, A., Knizhnerman, L., 2006. Two-and-half-dimensional forward and inverse modeling for marine CSEM problems, in: SEG Technical Program Expanded Abstracts, Society of Exploration Geophysicists. doi:10.1190/1.2370366.

Anderson, B.I., 2001. Modeling and Inversion Methods for the Interpretation of Resistivity Logging Tool Response. DUP Science, Delft, Netherlands.

Bakr, S.A., Pardo, D., Torres-Verdín, C., 2017. Fast inversion of logging-while-drilling resistivity measurements acquired in multiple wells. Geophysics 82, E111–E120. doi:10.1190/geo2016-0292.1.

Balay, S., Gropp, W.D., McInnes, L.C., Smith, B.F., 1997. Efficient management of parallelism in object oriented numerical software libraries, in: Arge, E., Bruaset, A.M., Langtangen, H.P. (Eds.), Modern Software Tools in Scientific Computing. Birkhäuser Press, pp. 163–202.

Buffa, A., Sangalli, G., Vázquez, R., 2010. Isogeometric analysis in electromagnetics: B-splines approximation. Computer Methods in Applied Mechanics and Engineering 199, 1143–1152. doi:10.1016/j.cma.2009.12.002.

Buffa, A., Sangalli, G., Vázquez, R., 2014. Isogeometric methods for computational electromagnetics: B-spline and T-spline discretizations. Journal of Computational Physics 257, 1291–1320. doi:10.1016/j.jcp.2013.08.015.

Calo, V.M., Pardo, D., Paszyński, M.R., 2011. Goal-oriented self-adaptive *hp* finite element simulation of 3D DC borehole resistivity simulations. Procedia Computer Science 4, 1485–1495. doi:10.1016/j.procs.2011.04.161.

Cangellaris, A.C., Wright, D.B., 1991. Analysis of the numerical error caused by the stair-stepped approximation of a conducting boundary in FDTD simulations of electromagnetic phenomena. IEEE Transactions on Antennas and Propagation 39, 1518–1525. doi:10.1109/8.97384.

Chaumont-Frelet, T., Nicaise, S., Pardo, D., 2018a. Finite element approximation of electromagnetic fields using nonfitting meshes for geophysics. SIAM Journal on Numerical Analysis 56, 2288–2321. doi:10.1137/16m1105566.

Chaumont-Frelet, T., Pardo, D., Rodríguez-Rozas, Á., 2018b. Finite element simulations of logging-while-drilling and extra-deep azimuthal resistivity measurements using non-fitting grids. Computational Geosciences 22, 1161–1174. doi:10.1007/s10596-018-9744-6.

Collier, N., Pardo, D., Dalcin, L., Paszynski, M., Calo, V.M., 2012. The cost of continuity: A study of the performance of isogeometric finite elements using direct solvers. Computer Methods in Applied Mechanics and Engineering 213-216, 353–361. doi:10.1016/j.cma.2011.11.002.

Cottrell, J.A., Hughes, T.J.R., Bazilevs, Y., 2009. Isogeometric Analysis: Toward Integration of CAD and FEA. John Wiley & Sons, Ltd, New York, NY.

Dagum, L., Menon, R., 1998. OpenMP: an industry standard API for shared-memory programming. IEEE Computational Science and Engineering 5, 46–55. doi:10.1109/99.660313.

Dalcin, L., Collier, N., Vignal, P., Côrtes, A.M.A., Calo, V.M., 2016. PetIGA: A framework for high-performance isogeometric analysis. Computer Methods in Applied Mechanics and Engineering 308, 151–181. doi:10.1016/j.cma.2016.05.011.

Davydycheva, S., 2011. Two triaxial induction tools: sensitivity to radial invasion profile. Geophysical Prospecting 59, 323–340. doi:10.1111/j.1365-2478.2010.00910.x.

Demkowicz, L., Monk, P., Vardapetyan, L., Rachowicz, W., 2000. De Rham diagram for *hp* finite element spaces. Computers & Mathematics with Applications 39, 29–38. doi:10.1016/s0898-1221(00)00062-6.

Garcia, D., Pardo, D., Calo, V.M., 2019. Refined isogeometric analysis for fluid mechanics and electromagnetics. Computer Methods in Applied Mechanics and Engineering 356, 598–628. doi:10.1016/j.cma.2019.06.011.

Garcia, D., Pardo, D., Dalcin, L., Paszyński, M., Collier, N., Calo, V.M., 2017. The value of continuity: Refined isogeometric analysis and fast direct solvers. Computer Methods in Applied Mechanics and Engineering 316, 586–605. doi:10.1016/j.cma.2016.08.017.

Gernez, S., Bouchedda, A., Gloaguen, E., Paradis, D., 2020. Aim4res, an open-source 2.5D finite differences MATLAB library for anisotropic electrical resistivity modeling. Computers & Geosciences 135, 104401. doi:10.1016/j.cageo.2019.104401.

Hageman, T., Fathima, K.M.P., de Borst, R., 2019. Isogeometric analysis of fracture propagation in saturated porous media due to a pressurised non-Newtonian fluid. Computers and Geotechnics 112, 272–283. doi:10.1016/j.compgeo.2019.04.030.

Hashemian, A., Pardo, D., Calo, V.M., 2021. Refined isogeometric analysis for generalized Hermitian eigenproblems. Computer Methods in Applied Mechanics and Engineering 381, 113823. doi:10.1016/j.cma.2021.113823.

Hughes, T.J.R., Cottrell, J.A., Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Computer Methods in Applied Mechanics and Engineering 194, 4135–4195. doi:10.1016/j.cma.2004.10.008.

Kaipio, J., Somersalo, E., 2007. Statistical inverse problems: Discretization, model reduction and inverse crimes. Journal of Computational and Applied Mathematics 198, 493–504. doi:10.1016/j.cam.2005.09.027.

Karypis, G., Kumar, V., 1998. A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM Journal on Scientific Computing 20, 359–392. doi:10.1137/s1064827595287997.

Kim, Y., Nakata, N., 2018. Geophysical inversion versus machine learning in inverse problems. The Leading Edge 37, 894–901. doi:10.1190/tle37120894.1.

Kushnir, D., Velker, N., Bondarenko, A., Dyatlov, G., Dashevsky, Y., 2018. Real-time simulation of deep azimuthal resistivity tool in 2D fault model using neural networks, in: SPE Annual Caspian Technical Conference and Exhibition. doi:10.2118/192573-ms.

Li, H., Wang, H., Wang, L., Zhou, X., 2020. A modified Boltzmann annealing differential evolution algorithm for inversion of directional resistivity logging-while-drilling measurements. Journal of Petroleum Science and Engineering 188, 106916. doi:10.1016/j.petrol.2020.106916.

Liu, C.R., 2017a. Theory of Electromagnetic Well Logging. Elsevier, Amsterdam, Netherlands.

Liu, H., 2017b. Principles and Applications of Well Logging. Springer, Berlin Heidelberg, Germany.

Liu, M., Grana, D., 2019. Accelerating geostatistical seismic inversion using TensorFlow: A heterogeneous distributed deep learning framework. Computers & Geosciences 124, 37–45. doi:10.1016/j.cageo.2018.12.007.

Ma, Z., Liu, D., Li, H., Gao, X., 2012. Numerical simulation of a multi-frequency resistivity logging-while-drilling tool using a highly accurate and adaptive higher-order finite element method. Advances in Applied Mathematics and Mechanics 4, 439453. doi:10.1017/S2070073300001739.

Nam, M.J., Pardo, D., Torres-Verdín, C., 2013. Simulation of borehole-eccentered triaxial induction measurements using a Fourier *hp* finite-element method. Geophysics 78, D41–D52. doi:10.1190/geo2011-0524.1.

Nguyen, D.M., Evgrafov, A., Gravesen, J., 2012. Isogeometric shape optimization for electromagnetic scattering problems. Progress In Electromagnetics Research B 45, 117–146. doi:10.2528/pierb12091308.

Nunes, C.M.B., Régis, C., 2020. GEMM3D: An edge finite element program for 3D modeling of electromagnetic fields and sensitivities for geophysical applications. Computers & Geosciences 139, 104477. doi:10.1016/j.cageo.2020.104477.

Pardo, D., Demkowicz, L., Torres-Verdín, C., Paszynski, M., 2006. Two-dimensional high-accuracy simulation of resistivity logging-while-drilling (LWD) measurements using a self-adaptive goal-oriented *hp* finite element method. SIAM Journal on Applied Mathematics 66, 2085–2106. doi:10.1137/050631732.

Pardo, D., Matuszyk, P.J., Puzyrev, V., Torres-Verdin, C., Nam, M.J., Calo, V.M., 2021. Modeling of Resistivity and Acoustic Borehole Logging Measurements Using Finite Element Methods. Elsevier Science.

Pardo, D., Torres-Verdín, C., 2015. Fast 1D inversion of logging-while-drilling resistivity measurements for improved estimation of formation resistivity in high-angle and horizontal wells. Geophysics 80, E111–E124. doi:10.1190/geo2014-0211.1.

Pardo, D., Torres-Verdín, C., Nam, M.J., Paszynski, M., Calo, V.M., 2008. Fourier series expansion in a non-orthogonal system of coordinates for the simulation of 3D alternating current borehole resistivity measurements. Computer Methods in Applied Mechanics and Engineering 197, 3836–3849. doi:10.1016/j.cma.2008.03.007.

Petra, C.G., Schenk, O., Anitescu, M., 2014a. Real-time stochastic optimization of complex energy systems on high-performance computers. Computing in Science & Engineering 16, 32–42. doi:10.1109/mcse.2014.53.

Petra, C.G., Schenk, O., Lubin, M., Gäertner, K., 2014b. An augmented incomplete factorization approach for computing the Schur complement in stochastic optimization. SIAM Journal on Scientific Computing 36, C139–C162. doi:10.1137/130908737.

Piegl, L., Tiller, W., 1997. The NURBS Book. 2nd ed., Springer-Verlag, New York, NY.

Rodríguez-Rozas, Á., Pardo, D., 2016. A priori Fourier analysis for 2.5D finite elements simulations of logging-while-drilling (LWD) resistivity measurements. Procedia Computer Science 80, 782–791. doi:10.1016/j.procs.2016.05.368.

Rodríguez-Rozas, Á., Pardo, D., Torres-Verdín, C., 2018. Fast 2.5D finite element simulations of borehole resistivity measurements. Computational Geosciences 22, 1271–1281. doi:10.1007/s10596-018-9751-7.

Sarmiento, A., Côrtes, A., Garcia, D., Dalcin, L., Collier, N., Calo, V., 2017. PetIGA-MF: A multi-field high-performance toolbox for structure-preserving B-splines spaces. Journal of Computational Science 18, 117–131. doi:10.1016/j.jocs.2016.09.010.

Schenk, O., Gärtner, K., 2004. Solving unsymmetric sparse systems of linear equations with PARDISO. Future Generation Computer Systems 20, 475–487. doi:10.1016/j.future.2003.07.011.

Schenk, O., Gärtner, K., Fichtner, W., 2000. Efficient sparse LU factorization with left-right looking strategy on shared memory multiprocessors. BIT Numerical Mathematics 40, 158–176. doi:10.1023/a:1022326604210.

Shahriari, M., Pardo, D., 2020. Borehole resistivity simulations of oil-water transition zones with a 1.5D numerical solver. Computational Geosciences 24, 1285–1299. doi:10.1007/s10596-020-09946-5.

Shahriari, M., Pardo, D., Moser, B., Sobieczky, F., 2020a. A deep neural network as surrogate model for forward simulation of borehole resistivity measurements. Procedia Manufacturing 42, 235–238. doi:10.1016/j.promfg.2020.02.075.

Shahriari, M., Pardo, D., Picon, A., Galdran, A., Del Ser, J., Torres-Verdín, C., 2020b. A deep learning approach to the inversion of borehole resistivity measurements. Computational Geosciences 24, 971–994. doi:10.1007/s10596-019-09859-y.

Shahrokhabadi, S., Cao, T.D., Vahedifard, F., 2019. Isogeometric analysis through Bézier extraction for thermo-hydro-mechanical modeling of saturated porous media. Computers and Geotechnics 107, 176–188. doi:10.1016/j.compgeo.2018.11.012.

Shen, J., Sun, W., 2008. 2.5-D modeling of cross-hole electromagnetic measurement by finite element method. Petroleum Science 5, 126–134. doi:10.1007/s12182-008-0020-6.

Shen, Q., Chen, J., Wu, X., Han, Z., Huang, Y., 2020. Parallel tempered trans-dimensional Bayesian inference for the inversion of ultra-deep directional logging-while-drilling resistivity measurements. Journal of Petroleum Science and Engineering 188, 106961. doi:10.1016/j.petrol.2020.106961.

Simona, A., Bonaventura, L., de Falco, C., Schöps, S., 2020. IsoGeometric approximations for electromagnetic problems in axisymmetric domains. Computer Methods in Applied Mechanics and Engineering 369, 113211. doi:10.1016/j.cma.2020.113211.

Simpson, R.N., Liu, Z., Vázquez, R., Evans, J.A., 2018. An isogeometric boundary element method for electromagnetic scattering with compatible B-spline discretizations. Journal of Computational Physics 362, 264–289. doi:10.1016/j.jcp.2018.01.025.

Tarantola, A., 2005. Inverse Problem Theory and Methods for Model Parameter Estimation. Society for Industrial and Applied Mathematics,

Philadelphia, PA.

Vogel, C.R., 2002. Computational Methods for Inverse Problems. Society for Industrial and Applied Mathematics, Philadelphia, PA.

Wang, H., Tao, G., Zhang, K., 2013. Wavefield simulation and analysis with the finite-element method for acoustic logging while drilling in horizontal and deviated wells. Geophysics 78, D525–D543. doi:10.1190/geo2012-0542.1.

Watzenig, D., 2007. Bayesian inference for inverse problems – statistical inversion. e & i Elektrotechnik und Informationstechnik 124, 240–247. doi:10.1007/s00502-007-0449-0.

Yang, F., Ma, J., 2019. Deep-learning inversion: A next-generation seismic velocity model building method. Geophysics 84, R583–R599. doi:10.1190/geo2018-0249.1.

Zhou, J., 2016. LWD/MWD Resistivity Tool Parameters. Maxwell Dynamics, Houston, TX.

Appendix A. Computer code availability

The computer codes contributed to the research results reported in the paper as well as the varying parameters and generated synthetic dataset are publicly available on https://gitlab.com/ali.hashemian/2.5d_em. The file `2.5D_EM_parallelrun.py` executes the parallel computations. We write this code in a Python 2.7 environment, which is based on Anaconda (available on https://repo.anaconda.com/archive). In this file, we set different parameters for our model problems and export them as inputs to the `run-riga-em.sh` file. This shell file consists of three parts. The preprocessing part that generates the computational mesh using Python. Therein we employ igakit (available on https://bitbucket.org/dalcinl/igakit) to create appropriate B-spline discretizations of the computational domain. The main part is a C code that solves the 2.5D borehole EM problem. The code needs PETSc (available on https://gitlab.com/petsc/petsc) and PetIGA (available on https://bitbucket.org/dalcinl/PetIGA) packages to be executed. It is important to note that PETSc should be configured with complex scalars, METIS, OpenMPI, ScaLAPACK, and Intel MKL libraries to have a computationally efficient solution. The latter library contains PARDISO as our sparse direct solver. Finally, the third part of the shell file performs the postprocessing in Python to calculate apparent resistivities based on attenuation ratio and phase difference.